What is Claimed is:

1. A method for verifying correctness of a component comprising:

receiving a number of randomly selected samples, M, $M \geq \frac{1}{\varepsilon}\ln\left(\frac{1}{\delta}\right)$,

wherein

δ represents a confidence value in the range $0 < \delta < 1$,

ε represents an accuracy level of p-est to its true value p, in the range $0 < \varepsilon < 1$,

p represents a probability that a randomly selected point is in accordance with a selected criterion, F,

p-est, an estimate of p based on M randomly selected samples, is zero,

and a probability that ($p \geq \varepsilon$) is equal to or less than δ;

determining if each of said randomly selected samples is not in accordance with said selected criterion, F; and

verifying correctness of said component based on said determining.

2. The method of Claim 1, further comprising:

determining that said component is not verified as correct if any one of said selected samples is in accordance with said selected criterion F; and

determining that said component is verified as correct if all of said selected samples are not in accordance with said selected criterion, F.

3. The method of Claim 2, wherein said samples are points, F uses a function f(x) where x is one of said points corresponding to one or more neural network inputs, f(x) is a neural network output for a corresponding one of said points, and said criterion F is that f(x) evaluates to a value that exceeds predetermined bounds.

4. The method of Claim 2, wherein said samples are points, F uses an error function e(x) represented as:

$$e(x) = f(x) - \varphi(x)$$

where x is one of said points corresponding to one or more neural network inputs,

f(x) is a neural network output for a corresponding one of said points,

$\varphi(x)$ is an expected output for a corresponding one of said points, and

wherein said criterion F is that e(x) evaluates to a value that exceeds predetermined bounds.

5. The method of Claim 2, wherein said component is one of: a neural network, a fuzzy logic model, a fuzzy logic classifier, and a statistical k-neighbor classifier.

6. The method of Claim 2, wherein said component is included in a system with at least one other component.

7. The method of Claim 6, wherein said component is a first component, and an output of a second component is used as an input to the first component, and the method further comprising:

determining whether said first component is verified as correct in accordance with error that may be introduced by said second component's output.

8. The method of Claim 6, wherein said component is a first component, and correctness of a second component is verified using said M samples, said second component producing an output which is an input to a third component, and the method further comprising:

determining if each of said randomly selected samples is in accordance with said selected criterion, F for said second component; and

determining that said second component is verified as correct unless a predetermined number, b, of said selected samples are in accordance with said selected criterion F, b being equal to or greater than 1.

9. The method of Claim 2, wherein said component is included in a system of an aircraft being evaluated in accordance with a certification.

10. The method of Claim 9, wherein δ and ε are both equal to or less than 10^{-6}.

11. The method of Claim 10, wherein ε is equal to or less than 10^{-9}.

12. A method for determining a number of randomly selected data values for verification of a component comprising:

receiving a value of zero for p-est, an estimate of p based on a random sample, p representing a probability that a randomly selected data value is in accordance with a selected criterion, F;

receiving δ representing a confidence value in the range $0< \delta< 1$;

receiving ε representing an accuracy level of p-est to its true value p, in the range $0< \varepsilon < 1$, wherein a probability that $(p \geq \varepsilon)$ is equal to or less than δ; and

determining said number of randomly selected data values, M, used for verification of a component wherein M is determined in accordance with p-est=0, δ, and ε.

13. The method of Claim 12, wherein said component is a neural network.

14. The method of Claim 13, wherein said neural network is a feed forward static neural network.

15. The method of Claim 12, wherein M is determined according to one of: a Bernoulli analysis based on p-est=0 and a Bayesian analysis based on p-est=0.

16. The method of Claim 15, wherein M is determined according to said Bernoulli analysis and M is $M \geq \frac{1}{\varepsilon}\ln\left(\frac{1}{\delta}\right)$.

17. The method of Claim 15, wherein M is determined according to said Bayesian analysis with a non-informative prior distribution, and M is $M \geq \frac{1}{\varepsilon}\ln\left(\frac{1}{\delta}\right)$-1.

18. The method of Claim 15, wherein M is determined according to said Bayesian analysis using a family of parameterized prior probability density functions, and M is $M \geq \frac{1}{\varepsilon}\ln\left(\frac{1+\delta}{\delta}\right)$-1.

19. A computer program product that verifies correctness of a component comprising code that:

receives a number of randomly selected samples, M, $M \geq \frac{1}{\varepsilon}\ln\left(\frac{1}{\delta}\right)$,

wherein

δ represents a confidence value in the range $0 < \delta < 1$,

ε represents an accuracy level of p-est to its true value p, in the range $0 < \varepsilon < 1$,

p represents a probability that a randomly selected point is in accordance with a selected criterion, F,

p-est, an estimate of p based on M randomly selected samples, is zero,

and a probability that ($p \geq \varepsilon$) is equal to or less than δ;

determines if each of said randomly selected samples is not in accordance with said selected criterion, F; and

verifies correctness of said component based on whether each of said randomly selected samples is in not in accordance with said selected criterion F.

20. The computer program product of Claim 19, further comprising code that:

determines that said component is not verified as correct if any one of said selected samples is in accordance with said selected criterion F; and

determines that said component is verified as correct if all of said selected samples are not in accordance with said selected criterion, F.

21. The computer program product of Claim 20, wherein said samples are points, F uses a function f(x) where x is one of said points corresponding to one or more neural network inputs, f(x) is a neural network output for a corresponding one of said points, and said criterion F is that f(x) evaluates to a value that exceeds predetermined bounds.

22. The computer program product of Claim 20, wherein said samples are points, F uses an error function e(x) represented as:

$$e(x) = f(x) - \varphi(x)$$

where x is one of said points corresponding to one or more neural network inputs,

f(x) is a neural network output for a corresponding one of said points,

$\varphi(x)$ is an expected output for a corresponding one of said points, and

wherein said criterion F is that e(x) evaluates to a value that exceeds predetermined bounds.

23. The computer program product of Claim 20, wherein said component is one of: a neural network, a fuzzy logic model, a fuzzy logic classifier, and a statistical k-neighbor classifier.

24. The computer program product of Claim 20, wherein said component is included in a system with at least one other component.

25. The computer program product of Claim 24, wherein said component is a first component, and an output of a second component is used as an input to the first component, and the computer program product further comprising code that:

determines whether said first component is verified as correct in accordance with error that may be introduced by said second component's output.

26. The computer program product of Claim 24, wherein said component is a first component, and correctness of a second component is verified using said M samples, said second component producing an output which is an input to a third component, and the computer program product further comprising code that:

determines if each of said randomly selected samples is in accordance with said selected criterion, F for said second component; and

determines that said second component is verified as correct unless a predetermined number, b, of said selected samples are in accordance with said selected criterion F, b being equal to or greater than 1.

27. The computer program product of Claim 20, wherein said component is included in a system of an aircraft being evaluated in accordance with a certification.

28. The computer program product of Claim 27, wherein δ and ε are both equal to or less than 10^{-6}.

29. The computer program product of Claim 28, wherein ε is equal to or less than 10^{-9}.

30. A computer program product that determines a number of randomly selected data values for verification of a component comprising code that:

receives a value of zero for p-est, an estimate of p based on a random sample, p representing a probability that a randomly selected data value is in accordance with a selected criterion, F;

receives δ representing a confidence value in the range $0< \delta< 1$;

receives ε representing an accuracy level of p-est to its true value p, in the range $0< \varepsilon < 1$, wherein a probability that $(p \geq \varepsilon)$ is equal to or less than δ; and

determines said number of randomly selected data values, M, used for verification of a component wherein M is determined in accordance with p-est=0, δ, and ε.

31. The computer program product of Claim 30, wherein said component is a neural network.

32. The computer program product of Claim 31, wherein said neural network is a feed forward static neural network.

33. The computer program product of Claim 30, wherein M is determined according to one of: a Bernoulli analysis based on p-est=0 and a Bayesian analysis based on p-est=0.

34. The computer program product of Claim 33, wherein M is determined according to said Bernoulli analysis and M is $M \geq \frac{1}{\varepsilon}\ln\left(\frac{1}{\delta}\right)$.

35. The computer program product of Claim 33, wherein M is determined according to said Bayesian analysis with a non-informative prior distribution, and M is $M \geq \frac{1}{\varepsilon}\ln\left(\frac{1}{\delta}\right)$-1.

36. The computer program product of Claim 33, wherein M is determined according to said Bayesian analysis using a family of parameterized prior probability density functions, and M is $M \geq \frac{1}{\varepsilon}\ln\left(\frac{1+\delta}{\delta}\right)$-1.